1 <u>TITLE OF THE INVENTION</u>

2 **Speed Converter for IEEE-1394 Serial Bus Network**

3 <u>BACKGROUND OF THE INVENTION</u>

4 <u>Field of the Invention</u>

5 The present invention relates generally to IEEE-1394 serial bus

6 networks, and more specifically to speed conversion of packets

7 transmitted between nodes attached to the IEEE-1394 serial buses.

8 <u>Description of the Related Art</u>

9 IEEE-1394 Serial Bus Standard supports high performance packet

10 transfer at speeds of 100 Mbps, 200 Mbps, and 400 Mbps on a high

11 reliability asynchronous transfer mode (though latency is not

12 guaranteed) and on a bandwidth guaranteed isochronous transfer

13 mode. The IEEE-1394 serial bus is best suited for digital video cameras.

14 With the built-in feature of IEEE-1394 interface, standardized digital

15 video cameras have met with wide reception among consumers.

16 In a single IEEE-1394 serial bus network, data transfer at different

17 speeds is possible. For example, a 100-Mbps transfer may proceed

18 between two nodes on a cable segment and a 400-Mbps transfer may

19 proceed between other two nodes on a separate cable segment. Since

20 high speed nodes support lower speed transfers, the same node can

21 communicate with a node at a low speed at one time and communicate

22 with another node at a higher speed at different time.

23 However, from the bandwidth savings viewpoint, the use of a low

24 speed node is not a favorable situation because the time taken to

25 transmit a given amount of information is longer than the time a higher

26 speed node takes to transmit the same amount of information. For

1   example, if a video channel is transmitted on a 100-Mbps isochronous

2   mode, a period of 40 microseconds is required during each 125-

3   microsecond cycle. Since the IEEE-1394 standard specifies that the

4   maximum amount of time available for isochronous transfer for each

5   cycle is 100 microseconds, the maximum number of video channels

6   which the current IEEE-1394 serial bus can support is only two.

7   Therefore, use of different speed nodes in a single IEEE-1394 serial bus

8   network represents a waste of otherwise usable bandwidth resource.

9   SUMMARY OF THE INVENTION

10   It is therefore an object of the present invention to reduce the

11   otherwise wasted bandwidth resource of an IEEE-1394 serial bus

12   network by providing a speed converter for converting the speed of

13   packets between nodes having different speed capabilities.

14   According to a first aspect of the present invention, there is

15   provided a speed converter for converting the speed of packets

16   transmitted between first and second communication nodes respectively

17   attached to first and second IEEE-1394 serial buses, comprising a first

18   transceiver node for receiving an inbound first packet at a first speed

19   from the first bus and transmitting an inbound second packet as an

20   outbound second packet at the first speed to the first bus, a second

21   transceiver node for transmitting the inbound first packet as an

22   outbound first packet at a second speed to the second bus and receiving

23   the inbound second packet at the second speed from the second bus, and

24   header translation circuitry for translating destination identifier of the

25   inbound first packet to destination identifier of the outbound first

26   packet according to a mapped correspondence between the first

1   transceiver node and the second communication node, and translating

2   destination identifier of the inbound second packet to destination

3   identifier of the outbound second packet.

4   According to a second aspect, the present invention provides a

5   speed converter for converting the speed of packets transmitted

6   between a plurality of first communication nodes attached to a first

7   IEEE-1394 serial bus and a plurality of second communication nodes

8   attached to a second IEEE-1394 serial bus. The speed converter includes

9   at least one first repeater node connected to the first bus, a first

10   transceiver node for receiving an inbound first asynchronous packet

11   from the first bus at a first speed via the at least one first repeater node

12   and transmitting an inbound second asynchronous packet as an

13   outbound second asynchronous packet at the first speed to the first bus

14   via the at least one first repeater node, the first transceiver node having

15   identifiers identifying the first transceiver node itself and the at least

16   one first repeater node, at least one second repeater node connected to

17   the second bus, a second transceiver node for transmitting the inbound

18   first asynchronous packet as an outbound first asynchronous packet to

19   the second bus at a second speed via at least one second repeater node

20   and receiving the inbound second asynchronous packet from the second

21   bus at the second speed via the at least one second repeater node and

22   receiving the inbound second asynchronous packet at the second speed

23   from the second bus via the at least one second repeater node, the

24   second transceiver node having identifiers identifying the second

25   transceiver node itself and the at least one second repeater node, and

26   header translation circuitry for translating destination identifier of the

1   inbound first asynchronous packet received by the first transceiver node

2   to destination identifier of the outbound first asynchronous packet

3   according to mapped relationships between the second communication

4   nodes and the first transceiver node and the at least one first repeater

5   node, and translating destination identifier of the inbound second

6   asynchronous packet received by the second transceiver node to

7   destination identifier of the outbound second asynchronous packet

8   according to mapped relationships between the first communication

9   nodes and the second transceiver node and the at least one second
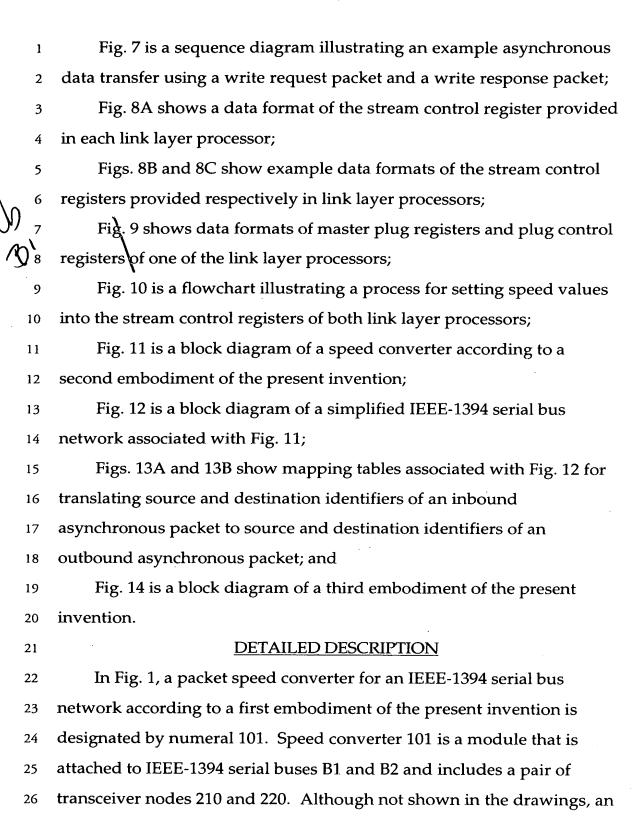
10   repeater node.

11   According to a third aspect of the present invention, the speed

12   converter is provided for converting the speed of packets transmitted

13   between a plurality of first communication nodes attached respectively

14   to a plurality of first IEEE-1394 serial buses and at least one second

15   communication node attached to a second bus. The speed converter

16   comprises a plurality of speed conversion units associated respectively

17   with the plurality of first buses. Each speed conversion unit includes a

18   first transceiver node for receiving an inbound first packet at a first

19   speed from the associated first bus and transmitting an inbound second

20   packet as an outbound second packet at the first speed to the associated

21   first bus, a second transceiver node for transmitting the inbound first

22   packet as an outbound first packet at a second speed to the second bus

23   and receiving the inbound second packet at the second speed from the ·

24   second bus, and header translation circuitry for translating destination

25 · identifier of the inbound first packet to destination identifier of the

26   outbound first packet according to mapped relationship between the

1   first communication node of the associated first bus and the at least one

2   second communication node, and translating destination identifier of

3   the inbound second packet to destination identifier of the outbound

4   second packet.

5   <div align="center">BRIEF DESCRIPTION OF THE DRAWINGS</div>

6   The present invention will be described in further detail with

7   reference to the accompanying drawings, in which:

8   Fig. 1 is a block diagram of a speed converter according to a first

9   embodiment of the present invention for converting the speed of packets

10   transmitted on the IEEE-1394 serial bus;

11   Fig. 2 is an illustration of speed setting values selected by the speed

12   setting switches of Fig. 1 for determining the speed of outgoing packets

13   of the speed converter;

14   Fig. 3 is a block diagram of a simplified IEEE-1394 serial bus

15   network useful for describing the speed conversion of primary packets

16   synchronized to cycle start packets;

17   Fig. 4 is a sequence diagram for illustrating incoming packets and

18   speed converted outgoing packets;

19   Fig. 5 is a block diagram of a simplified IEEE-1394 serial bus

20   network useful for describing mapping tables for mapping identifiers of

21   the network nodes;

22   Figs. 6A and 6B show mapping tables associated with Fig. 5 for

23   translating source and destination identifiers of an inbound

24   asynchronous packet to source and destination identifiers of an

25   outbound asynchronous packet;

1   Fig. 7 is a sequence diagram illustrating an example asynchronous

2   data transfer using a write request packet and a write response packet;

3   Fig. 8A shows a data format of the stream control register provided

4   in each link layer processor;

5   Figs. 8B and 8C show example data formats of the stream control

6   registers provided respectively in link layer processors;

7   Fig. 9 shows data formats of master plug registers and plug control

8   registers of one of the link layer processors;

9   Fig. 10 is a flowchart illustrating a process for setting speed values

10   into the stream control registers of both link layer processors;

11   Fig. 11 is a block diagram of a speed converter according to a

12   second embodiment of the present invention;

13   Fig. 12 is a block diagram of a simplified IEEE-1394 serial bus

14   network associated with Fig. 11;

15   Figs. 13A and 13B show mapping tables associated with Fig. 12 for

16   translating source and destination identifiers of an inbound

17   asynchronous packet to source and destination identifiers of an

18   outbound asynchronous packet; and

19   Fig. 14 is a block diagram of a third embodiment of the present

20   invention.

21   DETAILED DESCRIPTION

22   In Fig. 1, a packet speed converter for an IEEE-1394 serial bus

23   network according to a first embodiment of the present invention is

24   designated by numeral 101. Speed converter 101 is a module that is

25   attached to IEEE-1394 serial buses B1 and B2 and includes a pair of

26   transceiver nodes 210 and 220. Although not shown in the drawings, an

1   isochronous resource manager is attached to each of the buses B1 and B2

2   for channel number and bus bandwidth allocation for isochronous

3   transfers.

4   Transceiver node 210 includes a physical layer processor (LSI) 21

5   connected to the bus B1, a link layer processor (LSI) 31 and a speed

6   setting switch 41 for setting a desired first speed value into the link layer

7   processor 31. Likewise, the transceiver node 220 has a physical layer

8   processor 22 connected to the bus B2, a link layer processor 32 and a

9   speed setting switch 42 for setting a desired second speed value into the

10   link layer processor 32. Physical layer processors 21, 22 and the link

11   layer processors 31, 32 are designed to provide the functions specified by

12   the IEEE-1394 Standard. Further, each physical layer processor and the

13   associated link layer processor are interconnected via an interface

14   specified by the IEEE-1394 Standard. Note that speed setting may also

15   be achieved by coupling the speed setting switches 41 and 42 to the CPU

16   11 via the host bus S1 and setting desired speed values into the

17   respective link layer processors 31, 32 from the CPU 11.

18   Link layer processors 31 and 32 are connected to a host bus S1 and

19   are interconnected by an isochronous data path S2 and a sync signal

20   path S3 for transmission of synchronized isochronous packets. Host bus

21   S1 servers as a data path for asynchronous packets.

22   Each of the speed setting switches 41, 42 has a plurality of speed

23   setting values 0, 1, 2, 3, 4, 5 and 6 as shown in Fig. 2. Speed setting

24   values 0, 1, 2, 3, 4, 5 and 6 correspond to respective speed conversion

25   parameters. For speed setting values 0, 1, and 2, the speed of primary

26   packets (either isochronous or asynchronous) is converted to 100 Mbps,

1 200 Mbps and 400 Mbps, respectively. For speed setting values 3, 4 and

2 5, the same conversion speed values are used for isochronous packets,

3 but the speed of asynchronous packets is converted to a maximum

4 possible value. For speed setting value 6, the speed converter performs

5 no speed conversion so that packets are transmitted at the same speed

6 as they are received. Link layer processors 31 and 32 transmit primary

7 packets according to the speed value set by the associated speed setting

8 switches 41, 42.

9   To perform speed conversion for asynchronous transfers, the speed

10 converter performs header translation. For this purpose, the speed

11 converter further includes a central processing unit 11, a read-only

12 memory 12 and a random access memory 13, all of which are connected

13 to the host bus S1. To achieve packet header translation, the CPU 11

14 executes programmed instructions of the present invention stored in

15 the read-only memory 12. As will be described, the RAM 13 maintains

16 mapping tables which define relationships between old and new

17 destination identifiers. An asynchronous packet transmitted at a first

18 speed from the bus B1, for example, is received by the transceiver node

19 210 and temporarily stored in the RAM 13. One of these mapping tables

20 is used by the CPU 11 for translating the source and destination

21 identifiers contained in the stored packet header to the identifiers of the

22 node 220 and a destination node attached to the bus B2. The header-

23 translated packet is then transmitted from the transceiver node 220 to

24 the bus B2 at a second speed.

25   In addition, for speed conversion of asynchronous packets, the CPU

26 11 has the function of segmenting a high speed packet into a series of

1 low speed packets if the payload size of a transmitted high speed packet

2 exceeds the maximum payload size of the low speed packet, since the

3 maximum payload size of 400-Mbps packets is 2048 bytes while the

4 maximum payload size of 100 Mbps packets is 512 bytes.

5   On the other hand, channel number translation is performed for

6 isochronous (stream) transfers since the target node is identified by a

7 channel number instead of by a node identifier. For this reason, the

8 transceiver nodes 210 and 220 are respectively set to different channel

9 numbers before an isochronous transfer begins. As will be described in

10 detail later, stream packets transmitted on the bus B2, for example, are

11 received by the node 220 and passed through the isochronous data path

12 S3 to the node 210, where the channel number contained in their header

13 is translated to the channel number set in the node 210 and then

14 transmitted to the bus B1 at a second speed.

15   Note that the speed converter of this invention does not perform

16 transfer operations on all types of packets that propagate over the

17 associated buses. For example, all PHY packets transferred between

18 physical layers and all acknowledgment packets on the buses are not

19 transferred. The types of packets that are transferred through the speed

20 converter are the asynchronous packet and the stream packet which are

21 generally classified under the category of primary packets.

22   If two devices of different speeds are attached to the buses B1 and

23 B2 as represented by a 400-Mbps communication node 230 and a 100-

24 Mbps communication node 240 in Fig. 3, cycle start packets PS1 are sent

25 on bus B1 and cycle start packets PS2 are sent on bus B2, synchronized

26 to the cycle start packets PS1, as shown in Fig. 4. If a 400-Mbps primary

1 packet PA1 is sent from node 230 on bus B1 immediately following a

2 cycle start packet PS1-1, the packet is translated to a 100-Mbps primary

3 packet PA2 and forwarded onto bus B2 immediately following a cycle

4 start packet PS2-2. Likewise, if a 100-Mbps primary packet PB1 is

5 transmitted from node 240 on bus B2 immediately following a cycle

6 start packet PS2-3, the packet is translated to a 400-Mbps primary

7 packet PB2 and forwarded onto bus B1 immediately following a cycle

8 start packet PS1-4. In this way, the speed converter allows other high

9 speed packets to be multiplexed on the bus B1 to achieve efficient

10 utilization of the bus B1. More specifically, since 100-Mbps standard

11 digital video signals take some 40 microseconds per cycle to travel over

12 the IEEE-1394 serial bus, the current system can support only two

13 channels for simultaneous transmission. Therefore, the speed converter

14 of this invention can support eight channels of 100-Mbps digital video

15 signals by converting their speed to 400 Mbps.

16   All nodes of the network are identified by a node identifier which

17 consists of a 16-bit bus ID and a physical ID. Since the speed converter

18 of this invention may be connected in an existing bus which is assigned

19 a single bus identifier, the buses B1 and B2 are assumed to be assigned

20 the same bus identifier, "3FF$_h$", for example. Thus, in the present

21 invention, the physical ID can be used to represent the node ID of each

22 node of the network. For asynchronous transactions, a packet from a

23 sending node contains its node ID in the source address field of its

24 header and the node ID of a destination node in the destination address

25 field.

1   As described above, mapping tables are defined in the random

2   access memory 13 for translating the header of an inbound

3   asynchronous packet to the header of an outbound asynchronous

4   packet. As shown in Fig. 5, three communication nodes 231, 232, 233 are

5   attached to the bus B1 and a single communication node 241 is attached

6   to the bus B2. Assume that the communication nodes 231, 232 and 233

7   on bus B1 are assigned physical IDs "2", "1" and "0", respectively, and

8   the communication node 241 on bus B2 is assigned a physical ID "0".

9   Further, the transceiver nodes 210 and 220 of the speed converter are

10 assumed to be assigned physical IDs "3" and "1", respectively.

11 For a serial bus network such as shown in Fig. 5, two mapping

12 tables are defined: a mapping table 61 shown in Fig. 6A, and a mapping

13 table 62 shown in Figs. 6B. In the mapping table 61, the physical ID = 3

14 of transceiver node 210 on the side of bus B1 is mapped to the physical

15 ID = 0 of communication node 241 on bus B2. In the mapping table 62,

16 the physical ID = 1 of transceiver node 220 on the side of bus B2 is

17 mapped to the physical IDs = 2, 1 and 0 of communication nodes 231,

18 232 and 233 on bus B1.

19 When translating the header of an asynchronous packet

20 transmitted from any of the communication nodes 231, 232 and 233 to

21 the communication node 241, the CPU 11 uses the mapping table 616.

22 CPU 11 accesses the mapping table 62 to perform header translation on

23 asynchronous packets transmitted from the communication node 241 to

24 any of the communication nodes 231, 232 and 233.

25 Fig. 7 is a sequence diagram of asynchronous transactions between

26 communication nodes 231 and 241 when the speed setting switches 41

1   and 42 are respectively set to "2" (= 400 Mbps) and "0" (= 100 Mbps) (=

2   400 Mbps). At step SP1, the node 231 transmits a write request packet

3   at the set speed of 400 Mbps, with the source and destination fields

4   respectively set to the physical ID (= 2) of the source node 231 and

5   physical ID (= 3) of the transceiver node 210. Node 210 responds to the

6   write request packet with an ack_pending packet (step SP2). The write

7   request packet received by the node 210 is stored in the RAM 13. At step

8   SP3, the CPU 11 performs a header translation process by referencing

9   the mapping table 61 (Fig. 6A) to convert the source field of the packet

10 to the physical ID (= 1) of the transceiver node 220 on a predetermined

11 basis and convert the destination field to the physical ID (= 0) of node

12 241 according to the referenced mapping table 61. Subsequently, the

13 CPU 11 provides a header mapping process by identifying the write

14 transaction with a unique transaction label and mapping the old source

15 and destination IDs to the new source and destination IDs in the RAM

16 13. CPU 11 formulates a write request packet with a new header

17 containing the translated source and destination IDs and the transaction

18 label and forwards the packet to the link layer processor 32. Link layer

19 processor 32, knowing that the transmission speed is set equal to "0",

20 forwards the packet to the bus B2 at 100 Mbps (step SP4).

21   On receiving the write request packet from the bus B2, the

22 communication node 241 returns an ack_pending packet to the node

23 220 (step SP5). Then, the node 241 formulates a write response packet

24 with a header containing the transaction label and the source and

25 destination fields set to the physical ID (= 0) of its own node 241 and the

26 physical ID (= 1) of the transceiver node 220, respectively, and forwards

1   the packet onto the bus B2 at 100 Mbps (step SP6). The write response

2   packet is received by the transceiver node 220, which returns an

3   ack_complete packet to the node 241 (step SP7).

4   The write response packet from node 241 is received by the node

5   220 and stored in the RAM 13. CPU11 examines the RAM 13 by

6   comparing the transaction label and the source and destination IDs

7   contained in the write response packet with those stored in the RAM 13,

8   and knows that node 220 has received a corresponding write response

9   packet from node 241 in response to the write request packet which the

10   node 210 had previously received from node 231.

11   A header translation process proceeds in the CPU 11 by replacing

12   the contents of the source and destination fields of the packet header

13   with the physical ID (= 3) of node 210 and the physical ID (= 2) of node

14   231, respectively. The header-translated write response packet is read

15   out of the RAM 13 and passed to the link layer processor 31 of node 210

16   (step SP8). Since the transmission speed is set equal to "3", the link

17   layer processor 31 forwards the header-translated write response packet

18   to the bus B1 at 400 Mbps (step SP9). Node 231 receives this packet and

19   returns an ack_complete packet to the node 210 (step SP10).

20   If the communication node 241 initiates a transaction, the speed

21   converter 101 proceeds in the same manner as that described above

22   with the exception that the mapping table 62 (Fig. 6B) is used instead of

23   the mapping table 61.

24   Since most of low speed nodes currently available in the market

25   issue transaction requests only to an isochronous resource manager that

26   is attached to the same bus as the requesting node, the provision of only

1   one mapping table may be sufficient for such nodes. In the above

2   example, the communication node 241 is the low speed node. If the

3   node 241 is of the type of node that issues transaction requests only to

4   the isochronous resource manager connected to the bus B2, the mapping

5   table 62 is not required.

6   For stream packet transfers, each of the link layer processors 31

7   and 32 has a 32-bit stream control register (SCR) whose format is shown

8   in Fig. 8A. The stream control register is divided into seven fields. The

9   first 2-bit field is a "send/receive" field which is used to indicate

10   whether a stream packet is to be transmitted to a bus or received from

11   the bus. Specifically, decimal "1" and "2" in the send/receive field

12   indicates reception and transmission, respectively. The second 6-bit

13   "channel" field is used to specify the channel number allocated by the

14   isochronous resource manager to the stream packet. A "1" or a "0" in

15   the one-bit "i" field respectively indicate that the stream packet is an

16   isochronous stream packet or an asynchronous stream packet. The 3-bit

17   "speed" field indicates the transmission speed of the stream packet,

18   with decimal "0", "1" and "2" respectively indicating the speeds of 100,

19   200 and 400 Mbps. The 4-bit "overhead" field and the 14-bit "payload"

20   field are used to specify the bandwidth necessary for the transmission of

21   the stream packet. The 2-bit "reserved" field is a field that is reserved

22   for future use.

23   If the transceiver node 210 transmits a 400-Mbps isochronous

24   stream packet to the bus B1 and the transceiver node 220 receives a 100-

25   Mbps isochronous stream packet from the bus B2, and channel numbers

26   "3" and "63" are assigned to the nodes 210 and 220, respectively, the

1   stream control register of link layer processors 31 and 32 will be set as

2   shown in Figs. 8B and 8C. Specifically, in decimal notation, "2" and "1"

3   are set in the send/receive field of link layer processors 31 and 32, and

4   "3" and "63" are set in the respective channel fields, "2" and "0" are set

5   in the respective speed fields, and a "1" is set in the "i" fields. Arbitrary

6   values are shown set in the overhead and payload fields.

7   In the illustrated example, the channel number "63" of an inbound

8   isochronous stream packet from bus B2 is translated to the channel

9   number "3" for an outbound isochronous stream packet for

10   transmission to the bus B1. The process of setting different channel

11   numbers into the channel field of the SCR of each link layer processor

12   will be discussed later.

13   In a practical aspect of the present invention, the transceiver node

14   210 is provided with plug registers which are defined according to the

15   IEC-61883 Standard. Based on the parameters set in the plug control

16   registers, the settings of stream control register of the link layer

17   processor 31 are determined.

18   Specifically, as shown in Figs. 9A to 9D, four types of 32-bit

19   registers are provided: an output master plug register (oMPR), an output

20   plug control register (oPCR), an input master plug register (iMPR), and

21   an input plug control register (iPCR). The oMPR and oPCR are used for

22   setting the SCR of the link layer processor 31 for transmission of

23   isochronous packets and the iMPR and iPCR are used for setting the SCR

24   for reception of isochronous packets. Each MPR and the iPCR are

25   divided into six fields and the oPCR is divided into eight fields. The

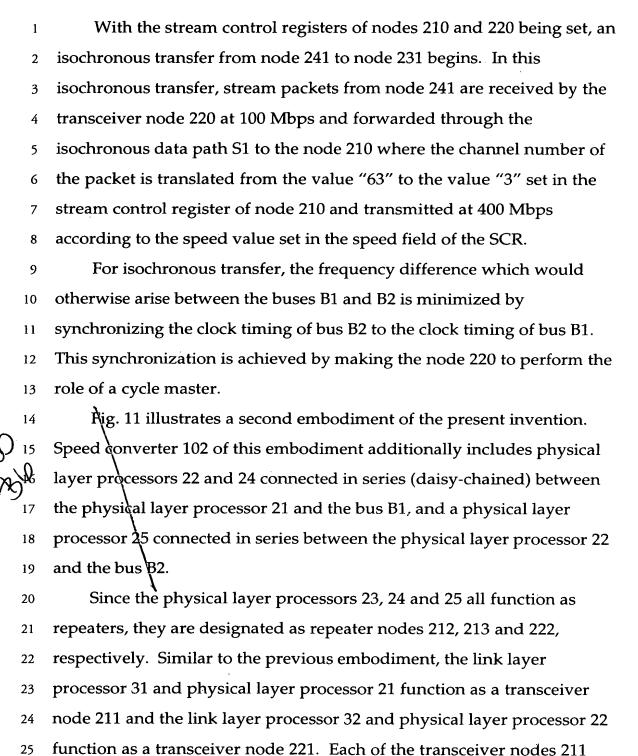26   initial values of these MPR and PCR registers in the transceiver node

1   210 are set equal to parameters set in the corresponding registers of the

2   communication node 231, for example, with the exception that the data

3   rate capability fields of both MPRs and the data rate field of the oPCR

4   are set equal to the speed setting value of switch 41. In the illustrated

5   examples of Figs. 8B and 8C, these data rate capability and data rate

6   fields of node 210 are set equal to the speed value of 400 Mbps.

7   In operation, the transceiver node 211 translates a first channel

8   number contained in an isochronous packet from the bus B2 to a second

9   channel number set in the channel number field of the oPCR when a

10   value indicating transmission of an isochronous packet is set in the

11   oMPR. The transceiver node 211 translates the second channel number

12   contained in an isochronous packet from the bus B1 to the first channel

13   number contained in the isochronous packet received from the bus B2

14   when a value indicating reception of an isochronous packet is set in the

15   iPCR.

16   According to the IEEE-1394 Standard, each node of the network has

17   a configuration ROM in which the capability and functions of the node

18   are stored. Assume that the communication node 231 initiates an

19   isochronous transaction by transmitting a read request packet to the

20   node 210 in the same manner as described above in connection with

21   asynchronous transfers in order to know what functions the node 210

22   are capable of. In response to the read request packet from node 231, the

23   transceiver node 210 accesses its own configuration ROM to read the

24   functions of node 210. After header translation, the data read from the

25   configuration ROM are set into the payload field of the read request

26   packet and this header-translated packet is forwarded from node 220 to

1 node 241. In response, the node 241 accesses its own configuration ROM

2 to read its contents and returns a read response packet containing the

3 contents of the configuration ROM of node 241. After header

4 translation, the node 210 transmits the read response packet back to the

5 requesting node 231. Node 231 examines the contents of the read

6 response packet and determines the target node that can provide the

7 capability which is desired by the requesting node. Note that the

8 configuration ROM just described above is preferably in an address

9 space from "FFFF F000 0400" to "FFFF F000 07FC" defined on the

10 address space of each of the buses B1 and B2.

11   After determining the target node, different channel numbers are

12 set in the stream control registers of link layer processors 31 and 32

13 according to a flowchart shown in Fig. 10.

14   At step 301, the node 231 acquires a channel number (i.e., "3") from

15 the isochronous resource manager that is attached to the bus B1. Node

16 231 initiates a lock transaction to the node 210 by setting the acquired

17 channel number into the channel number field of the oPCR with and a

18 "1" into the point-to-point connection counter field of the oPCR (step

19 302). At step 303, the node 210 sets the send/receive field and channel

20 field of its own stream control register with values "2" and "3",

21 respectively. Thus, the node 210 is set in a transmit mode for

22 transmitting a stream packet of channel number "3" to the node 231. At

23 step 304, the node 220 sets a value of "1" in the send/receive field of its

24 own stream control register and a default value of "63" in its channel

25 field.

1   With the stream control registers of nodes 210 and 220 being set, an

2   isochronous transfer from node 241 to node 231 begins. In this

3   isochronous transfer, stream packets from node 241 are received by the

4   transceiver node 220 at 100 Mbps and forwarded through the

5   isochronous data path S1 to the node 210 where the channel number of

6   the packet is translated from the value "63" to the value "3" set in the

7   stream control register of node 210 and transmitted at 400 Mbps

8   according to the speed value set in the speed field of the SCR.

9   For isochronous transfer, the frequency difference which would

10 otherwise arise between the buses B1 and B2 is minimized by

11 synchronizing the clock timing of bus B2 to the clock timing of bus B1.

12 This synchronization is achieved by making the node 220 to perform the

13 role of a cycle master.

14   Fig. 11 illustrates a second embodiment of the present invention.

15 Speed converter 102 of this embodiment additionally includes physical

16 layer processors 22 and 24 connected in series (daisy-chained) between

17 the physical layer processor 21 and the bus B1, and a physical layer

18 processor 25 connected in series between the physical layer processor 22

19 and the bus B2.

20   Since the physical layer processors 23, 24 and 25 all function as

21 repeaters, they are designated as repeater nodes 212, 213 and 222,

22 respectively. Similar to the previous embodiment, the link layer

23 processor 31 and physical layer processor 21 function as a transceiver

24 node 211 and the link layer processor 32 and physical layer processor 22

25 function as a transceiver node 221. Each of the transceiver nodes 211

1   and 221 further consists of a software-implemented transaction layer.
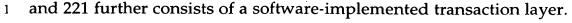
2   All nodes of the network are identified by a physical ID.

3   As shown in Fig. 12, communication nodes 311 and 312 are

4   attached to the bus B1 and communication nodes 321, 322 and 323 are

5   attached to the bus B2. For illustration, the nodes 311 and 321 are

6   assumed to be a digital video camera with a transmission speed of 200

7   Mbps, while the other nodes are personal computers capable of

8   operating at 400 Mbps.

9   Note that the link layer processor 31 of transceiver node 211 is

10 capable of receiving asynchronous packets from bus B1, containing not

11 only its own physical ID but also the physical IDs of the repeater nodes

12 212 and 213. Likewise, the link layer processor 32 of transceiver node

13 221 is capable of receiving asynchronous packets from bus B2,

14 containing the physical ID of the repeater node 222 as well as its own

15 physical ID. The speed of transmission of asynchronous packets from

16 each of the transceiver nodes 211 and 221 is set to the maximum

17 available value and the speed of transmission of stream packets is set to

18 400 Mbps.

19   Figs. 13A and 13B show two mapping tables 71 and 72 defined in

20 the RAM 13. In the mapping table 71, the transceiver node 211 and

21 repeater nodes 212, 213 are mapped to the communication nodes 321,

22 322 and 323 on the bus B2, respectively. In the mapping table 72, the

23 transceiver node 221 and repeater node 222 are mapped to the

24 communication nodes 311 and 312 on the bus B1, respectively.

25   If the transceiver node 211, for example, receives a configuration-

26 ROM read request packet from communication node 312 on bus B1.

1   Configuration ROM data of all communication nodes are stored in the

2   RAM 13. In response to the read request from node 312, the transceiver

3   node 211 reads from the RAM 13 the configuration ROM data of

4   communication node 321 on bus B2 that is defined in the mapping table

5   71 as a node corresponding to the transceiver node 211 and returns a

6   read response packet containing the read configuration ROM data.

7   Therefore, the communication nodes 321, 322, 323 on bus B2 are

8   "visible" from all communication nodes on bus B1, instead of the nodes

9   211, 212 and 213. Likewise, the communication nodes 311 and 312 on

10 bus B1 are "visible" from all communication nodes on bus B2, instead of

11 the nodes 221 and 222.

12   If the computer node 312 on bus B1 performs a configuration-ROM

13 read request transaction on the other nodes of bus B1, it will recognize

14 nodes 311 and 211 as a digital video camera. Likewise, if the computer

15 nodes 322 and 323 on bus B2 perform a configuration-ROM read request

16 transaction on the other nodes of bus B2, they will recognize nodes 321

17 and 221 as a digital video camera.

18   Following the configuration-ROM read request transaction, the

19 computer node 312 on bus B1 sends an asynchronous request packet to

20 the transceiver node 211. Specifically, the computer node 312 specifies

21 the allocated channel number and the set speed by performing a write

22 transaction on a register whose location is offset by "$60C_h$" from the

23 reference address value and starts a data transfer by performing a write

24 transaction on a register whose offset value is "614_h". Note that the

25 reference address value is written on the Unit Dependent Directory of

26 the configuration ROM of node 312.

1   Transceiver node 211, on receiving the write request packet, stores

2   the packet in the RAM 13. CPU 11 performs a header translation by

3   rewriting the destination field of the request packet (which contains the

4   physical ID of node 211) with the physical ID of node 321 according to

5   the mapping table 71 and rewriting its source field (which contains the

6   physical ID of node 312) with the physical ID of node 222 according to

7   the mapping table 72.

8   The header-translated write request packet is then forwarded from

9   the RAM 13 to the transceiver node 221 and then transmitted at the

10   maximum speed of 200 Mbps to the digital video camera 321 via the

11   repeater node 222.

12   Digital video camera 321 responds to the write request packet with

13   a write response packet, which is received by the transceiver node 221

14   via the repeater node 222 and stored in the RAM 13 for header

15   translation. CPU 11 performs this header translation by rewriting the

16   destination field of the response packet (which contains the physical ID

17   of node 222) with the physical ID of node 312 according to the mapping

18   table 72 and rewriting its source field (which contains the physical ID of

19   node 321) with the physical ID of node 211 according to the mapping

20   table 71. The header-translated write response packet is forwarded

21   from the RAM 13 to the transceiver node 211 and then transmitted to

22   the computer node 312 at 400 Mbps which is the maximum transfer

23   speed between the nodes 211 and 312.

24   When the write transaction is successful, the speed converter 102

25   proceeds to set the stream control registers of the respective link layer

26   processors 31, 32 so that isochronous packets from the digital video

1 camera 321 can be forwarded through the transceiver node 211 onto the

2 bus B1 at 400 Mbps.

3   In the IEEE-1394 serial bus network, bus reset is initiated under

4 various circumstances, which forces all nodes into their initialization

5 state, thereby initiating the configuration process. Preferably, the

6 transceiver node 221 is provided with a bus reset recovery feature to

7 minimize the interruption of data transfer caused by a bus reset. If bus

8 reset occurs on the bus B2 during the data transfer from the digital video

9 camera 321 to the transceiver node 221, the latter senses this condition

10 and performs a write transaction on the register of offset address value

11 "614_h" to enable the video camera 321 to reinitiate the isochronous

12 transfer by resetting the transmit/receive state of its stream control

13 register.

14   Prior to storage of the configuration ROM data of all

15 communication nodes of the network into the random access memory

16 13, the lower 64 bits of Bus_Info_Block of the configuration data and the

17 lower 64 bits of Node_Unique_Id leaf are preferably rewritten with the

18 EUI-64 values (Extended Unique Identifier, 64 bits) and the

19 module_vendor_id field of the Module_Vendor_Id entry is rewritten

20 with the company ID indicating the manufacturer of the speed

21 converter 102. The EUI-64, consisting of a 24-bit manufacturer's

22 identifier and a 40-bit chip identifier, is an identifier which is assigned

23 uniquely to all nodes of the network which are provided with the

24 general format configuration ROM.

25   When a configuration ROM read request packet is asserted on a

26 given transceiver node of speed converter 102, the rewritten

1 configuration data is read from the RAM 13 and only the device

2 function is enabled to appear as if it were the same entity as the node

3 that corresponds in the mapping table to the given transceiver node. In

4 addition, the rewriting of the configuration ROM data eliminates the

5 need to alter the specifications of digital video controllers.

6   More specifically, if the transceiver node 211 receives a

7 configuration ROM read request packet from the bus B1, it reads

8 configuration ROM data from the memory 13 corresponding to the

9 destination identifier contained in the received read request packet and

10 transmits a read response packet to the bus B1 containing the read

11 configuration ROM data. If the transceiver node 221 receives a

12 configuration ROM read request packet from the bus B2, it reads

13 configuration ROM data from the memory 13 corresponding to the

14 destination identifier contained in the received read request packet and

15 transmits a read response packet to the second bus containing the read

16 configuration ROM data.

17   A third embodiment of the present invention is shown in Fig. 14 in

18 which speed converters 101A, 101B and 101C of the same configuration

19 as that of Fig. 1 are incorporated in a single speed converter 103. These

20 speed converters are set at different speed values. The transceiver nodes

21 210 of the speed converters 101A, 101B, 101C are connected to buses B1-

22 1, B1-2 and B1-3, respectively, and the transceiver nodes 220 of these

23 speed converters are connected in series to the bus B2. High speed

24 communication nodes 401, 402 and 403 are attached to the buses B1-1,

25 B1-2 and B1-3, respectively, and low speed communication nodes 404

26 and 405 are attached to the bus B2. Similar to the first embodiment, all

1   the transceiver nodes 220 are recognized by the communication nodes

2   404 and 405 as if they were the high speed communication nodes 401,

3   402 and 403. In this way, data transfer can be provided at a number of

4   different speeds.

5   As described above, in the speed converter for an IEEE-1394 serial

6   bus network, a first transceiver node represents a low speed

7   communication node on one bus and performs data transfer with a high

8   speed communication node on the other bus and a second transceiver

9   represents the high speed communication node and performs data

10   transfer with the low speed communication node. A high speed device

11   can maintain its transmission speed when communicating with a low

12   speed device through the speed converter of the present invention, and

13   a substantial resource saving is achieved for the IEEE-1394 serial bus.

14   Experiments showed that more than three digital video channels were

15   successfully transmitted on the same IEEE-1394 serial bus.